

(19) World Intellectual Property Organization
International Bureau(43) International Publication Date
20 September 2001 (20.09.2001)

PCT

(10) International Publication Number
WO 01/69488 A1(51) International Patent Classification⁶: G06F 17/60

(21) International Application Number: PCT/US01/07507

(22) International Filing Date: 9 March 2001 (09.03.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/188,551 10 March 2000 (10.03.2000) US

(71) Applicant and

(72) Inventor: JONES, Charles, P. [US/US]; 34625 S. Hadwag Creek Rd., Leesburg, FL (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): FIERING, Kenneth, B. [US/US]; 774 Mays Boulevard, #10, Incline Village, NV 89451 (US). DESIENO, Duane [US/US]; San Diego, CA (US).

(74) Agents: WOYCECHOWSKY, David, B. et al.; Suite 2600, 600 West Broadway, San Diego, CA 92101 (US).

(81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

- with international search report
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: VEHICLE SCHEDULING SYSTEM

(57) Abstract: A delivery vehicle scheduling system creates a route for one or more vehicles to effectuate efficient deliveries. Parameters related to the recipients of the deliveries (10) as well as to the vehicles and costs of late and/or early arrival (20) are used to analyze available time slots and to select those that have a lowest-cost impact on the current delivery schedule. An optimization routine takes newly generated time slots and inserts them into the existing delivery schedule (801). Optimization is performed by a neural network or genetic algorithm that manipulates the legs of each vehicle's route such the aggregate cost of the entire delivery schedule for all vehicles is minimized (40). Time slots for subsequent customers are generated based on this newly optimized schedule, and those new time slots are again optimized.

WO 01/69488 A1

VEHICLE SCHEDULING SYSTEM

5

PRIORITY INFORMATION

The present application claims all applicable priority and other legal benefit based on the following application(s): U.S. provisional application number

10 60/188,551, filed March 10, 2000.

FIELD OF THE INVENTION

15 The present invention relates to a system for automated scheduling of the destination(s) and arrival time(s) for vehicles, such as delivery trucks.

BACKGROUND OF THE INVENTION

Having goods delivered to one's home or business is a convenience that often
20 comes with a steep price. As anyone who has ever had furniture or other goods delivered can attest, an enormous time window, typically on the order of four hours, e.g. "between noon and 4 p.m.," is assigned to the truck that will deliver those goods. Agreeing to such a delivery often requires the customer to make arrangements, such as leaving work early, in order to be at home during that window, until the goods
25 arrive. Yet despite the generous expanse of the time window, the truck inevitably arrives late. And often not a few minutes late, but rather hours late. This can be the

result of traffic delays or, more typically, a poorly configured delivery route with all too many stops. The customer usually regards this as very poor service.

These problems arise because the delivery scheduling system must try to simultaneously optimize two conflicting goals, good customer service and low
5 cost/high efficiency. Some of the features associated with good customer service are short delivery time windows, a wide variety of available delivery times, the ability to select the delivery time when the customer places his order, and prompt arrival of the delivery vehicle. High efficiency for the delivering company requires delivering to as many customers as possible, with as few vehicles as possible in the delivery shift and
10 is usually achieved by sacrificing the requirements listed above for good customer service.

Some solutions have attempted to improve the efficiency of delivery schedules, but those attempts have been limited to software that chooses the roads that will decrease travel time. A delivery truck is still assigned a fixed route, and any
15 customer lying on that route is assigned to that truck. Only after all of the truck's destinations have been registered will the street-selection software churn out the directions that the truck driver should follow in order to arrive at successive destinations as quickly as possible.

A solution that allows the delivering company to optimize both efficiency and
20 customer service and precisely control the tradeoff is highly desirable. Furthermore, the ability to operate fully automatically is obviously of great benefit, particularly in Internet-based applications.

BRIEF SUMMARY OF THE PRESENT INVENTION

A need exists for an improved way to schedule the delivery of goods. The present invention is directed, in general, to a system for scheduling a plurality of destinations and arrival times for vehicles such as delivery trucks. One technique to optimize two conflicting goals is utilization of a fuzzy determination function. Fuzzy refers to the fact that the function used is smoothly varying as opposed to a step function. The preferred embodiment of the present invention uses this technique to maximize efficiency while still delivering excellent customer service. It allows a delivery time to be assigned when the customer orders, i.e. in real time, and all the delivery parameters such as delivery window, maximum delivery distance and promptness of arrival are directly controllable through the form of the fuzzy determination function. Efficient scheduling is obtained by checking, at the time the customer orders, all available delivery times that are feasible for a vehicle to reach the customer. The software then selects several of the most cost-effective times, spread out over the day or part of the day as chosen by the customer, e.g. morning, and offers those choices to the customer. In one embodiment, for example, the customer first selects morning, afternoon, evening, or night, each a 3 or 4 hour segment, and the software will then suggest times about an hour apart if they are available. This technique makes use of the fact that what most customers care about is having a fixed time and not a long window over which they must wait. Efficiency is achieved because the suggested times are the best ones for the delivering company. A rule-based system can also be employed to define the cost, however, the fuzzy determination system usually is more effective.

Further efficiency is achieved by continually optimizing the schedule by shifting deliveries between trucks. The "cost" or "efficiency" of a given set of

assignments of deliveries to vehicles is calculated using the fuzzy determination function and then a neural network or genetic algorithm performs the optimization over the possible sets of assignments. The "cost" can include actual expenses such as fuel and labor as well as early and late arrival penalties as assigned by the delivering
5 company. The definition of these costs controls the allowable window of delivery, maximum distance allowable, etc. The neural network or genetic algorithm optimization is extremely fast and enables full optimization in real time. This is very important because, in general, the number of calculations required increases as n factorial, where n is the number of delivery stops.

10 These artificial intelligence techniques can operate in a fully automatic manner when a simple interface to the customer is included in the system. By "automatic", we mean without the need for human intervention. In many embodiments of the present invention, the Internet provides an excellent platform for automated scheduling in real time. While we are emphasizing this real time aspect because of
15 the importance of Internet applications, the present invention can also provide highly efficient automated scheduling solutions in older "after-the-fact" delivery schemes. In fact, if one eliminates early and late penalties, for instance, or defines the cost determination function to be broad, e.g. several hours, the system will give very efficient traditional minimum distance solutions after all customers have been
20 determined and can do so automatically. These solutions in general, however, lie at the poor customer service end of a continuum of customer service versus efficiency. There are many applications, nonetheless, for which this is appropriate and the present invention offers an automatic, fast, and efficient means for scheduling.

The preferred embodiments also allow for multiple types of reservations in the
25 schedule. The first type of reservation is the "usual" or "normal" reservation that is

added to the schedule at the time the customer decides to order. Once inserted it is has a fixed target arrival time and will not be removed unless the customer or authorized user specifically directs the program to do so. The second type are 'hard-wired' reservations. This category allows for assigning a set delivery time to a customer and having it reappear at regular intervals. For example, a customer might wish a product delivered on Mondays at 12:00 pm every week. Hard-wired reservations, like normal reservations are not removed from the schedule once inserted unless the program is specifically directed to do so. The third type of reservation is referred to as a "Ghost Reservation". The Ghost Reservation is a vehicle for accomplishing two things. First, it can be used to give the program statistical information known ahead of time to increase delivery efficiency. If, for instance, a company keeps accurate records of delivery requests on a stop-by-stop basis, statistical predictions of times and locations can be preloaded into the schedule by assigning ghost stops. When the time slot generation routine checks all available times, it is allowed to replace any of the ghost reservations with a normal (real as opposed to ghost) reservation. The easiest way to understand this effect is to think of an isolated neighborhood that always gets numerous deliveries within some time period of the day. After the first person from that neighborhood makes a reservation, the time slot routine may send the truck to a location further away from that neighborhood than is desirable before other customers from the area make their reservations. With ghosts in the schedule for the neighborhood, however, the time slot routine and optimizer routine treat the ghosts as "normal" reservations while determining the cost. The program thus knows via the cost function that it should not venture far from the isolated neighborhood because it must return later. This leads to the second purpose of ghosts, enabling delivery to a remote location. Again

thinking of our remote neighborhood, it is possible that the program may not deliver to the neighborhood at all because the cost of driving there exceeds the maximum cost set by the user on an individual delivery basis, even though the deliveries may be cost effective when considered in aggregate. Defining ghosts overcomes this problem. In
5 practice, the neighborhood in question must truly be remote to the normal delivery area for the scheduler to have any difficulty in efficiently handling it without Ghosts.

Accordingly, it is an object of the present invention to provide an automatic system that schedules, in real time if desired, the travel routes of one or more vehicles to maximize their efficiency and minimize cost in visiting prescribed destinations on-
10 time.

It is another object of the present invention to create delivery schedules for a plurality of delivery vehicles such as delivery trucks, which schedules are based on collective cost minimization for the system subject to constraints that allow a predefined level of customer service.

15 These and other objects may be achieved by providing a method for scheduling travel of one or more vehicles, comprising (a) selecting a time slot from among a plurality of time slots, the selected time slot designating a time of vehicle arrival; (b) incorporating the selected time slot into an existing travel schedule for one or more vehicles, and (c) rearranging the assignments in real time or in batch
20 processes to provide greater efficiency.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will become more fully understood from the detailed description given below, together with the accompanying drawings that are given by

way of illustration only and thus are not to be construed as limiting the scope of the present invention. In the drawings:

Fig. 1 shows a sample dialog box that helps a user configure parameters in an embodiment of the present invention.

5 Fig. 2 shows a sample graph of a cost function generated according to an embodiment of the present invention.

Fig. 3 shows a sample graph of a cost function generated according to another embodiment of the present invention.

Fig. 4 shows a sample graph of a cost function generated according to still
10 another embodiment of the present invention.

Fig. 5 shows a sample graphic of a delivery route generated according to an embodiment of the present invention.

Fig. 6 shows a table of information displayed to a user according to an embodiment of the present invention.

15 Fig. 7 shows another table of information displayed to a user according to an embodiment of the present invention.

Fig. 8 shows a sample dialog box useful in assisting the user to enter information according to an embodiment of the present invention.

Fig. 9 shows a sample dialog box useful in assisting the user to enter
20 additional information according to an embodiment of the present invention.

Fig. 10 shows a table of information generated according to an embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Before starting a description of the Figures, some terms will now be defined.

DEFINITIONS

the present invention: means at least some embodiments of the present
5 invention; references to various feature(s) of the "present invention" throughout this document do not mean that all claimed embodiments or methods include the referenced feature(s).

location-time: proposed or actual location and associated time preference for
a visit by a vehicle; the location-time may also include cargo data specifying cargo
10 (human, animal or other) to be transferred to and/or from the vehicle during the visit; the time preference may take the form of a specific time, a set of discrete times, a range of times aor a combination of these types;

efficacy value: a number or other code reflecting the overall desirability of a
proposed schedule; the efficacy value preferably reflects all qualities that impact
15 overall desirability of a schedule; efficacy considerations may include, but are not limited to, cost considerations, efficiency considerations, time considerations, timeliness considerations, customer satisfaction, vehicle operator satisfaction, compliance with labor laws, and energy cost considerations.

vehicle: includes, but is not limited to, land vehicles, sea vehicles,
20 undersea air vehicles, space vehicles and underground vehicles.

travel: includes, but is not limited to, land travel, sea travel, undersea travel, space travel and underground travel.

To the extent that the definitions provided above are consistent with ordinary,
plain and accustomed meanings (as generally evidenced, inter alia, by dictionaries
25 and/or technical lexicons), the above definitions shall be considered supplemental in

nature. To the extent that the definitions provided above are inconsistent with ordinary, plain and accustomed meanings (as generally evidenced, inter alia, by dictionaries and/or technical lexicons), the above definitions shall control. If the definitions provided above are broader than the ordinary, plain and accustomed meanings in some aspect, then the above definitions will control at least in relation to their broader aspects.

To the extent that a patentee may act as its own lexicographer under applicable law, it is hereby further directed that all words appearing in the claims section, except for the above-defined words, shall take on their ordinary, plain and accustomed meanings (as generally evidenced, inter alia, by dictionaries and/or technical lexicons), and shall not be considered to be specially defined in this specification. Notwithstanding this limitation on the inference of "special definitions," the specification may be used to evidence the appropriate ordinary, plain and accustomed meanings (as generally evidenced, inter alia, by dictionaries and/or technical lexicons), in the situation where a word or term used in the claims has more than one alternative ordinary, plain and accustomed meaning and the specification is helpful in choosing between the alternatives.

One objective is to build and optimize a delivery schedule that a delivery vehicle uses in order to meet all of its delivery requirements. Although the present invention finds particular applicability for use with delivery services, it will become apparent upon reviewing this description that it does not require that goods actually be left behind by the truck or other traveling vehicle. Accordingly, the terms "delivery" and "delivery vehicle" are intended to refer broadly to any vehicle, which can take one of several paths to one or more destinations, regardless whether they are traveling there to deliver goods and services, or for any other reason. Similarly, the present

invention will find applicability to vehicles other than land vehicles, such as boats and aircraft, and/or to a system that employs a mixture of such vehicles.

The present invention is comprised of software routines and an architecture that applies certain artificial intelligence (AI) techniques in a unique way to generate a
5 delivery schedule that one or more vehicles should follow in order to arrive at prescribed destinations in a most efficient manner. When said software routines and AI techniques are applied to accompanying simulations, they also provide a useful tool for understanding the impact of constraints on the ability to successfully deliver orders to customers. For example, the question of how many trucks are needed to
10 satisfy a given number of customers in a day can be answered using hypothetical situations in the automatic schedule building embodiments of the present invention. For real customers, it is possible to determine the average delivery cost for a customer and discover the effects of distance from a warehouse on the likelihood of obtaining a delivery time slot. The system can simulate the real world with high fidelity because
15 of the automated nature of the software. One merely needs to assign random or probabilistic selections of the suggested delivery times by the fictitious customers, and the system performs the scheduling as it would in the real world application. The scheduling program, in both real world and simulation applications, preferably includes a neural network or genetic algorithm to optimize the assignment of
20 deliveries to vehicles. The simulation can be operated with the neural net optimization performed after scheduling a number of customers, to minimize computer time, or simultaneously with the addition of customers. Additional features include the ability to save optimized schedules and to load saved schedules rather than re-calculate them.

There are two fundamental routines that are used in the preferred embodiments. First, a time slot generation routine decides where to insert a new destination (customer) into an existing schedule. Obviously, the schedule is initially empty, although predetermined starting, end, and even intermediate locations may be designated. The time slot generation routine ranks the available time slots according to the respective increases in total delivery cost that they would incur, and selects, either with or without a customer's or other user's input, the best (lowest cost) slots.

The second routine is a schedule optimization routine, which takes the existing schedule and looks for ways to rearrange the appointments on the vehicles. The optimization routine differs from the time slot generation routine by its ability to change the scheduled delivery route(s) of the delivery vehicle(s), such as by inserting a new leg into the route, or by re-designating one destination to a different delivery vehicle so that, in aggregate, the total cost is minimized, or by otherwise manipulating the schedule.

In application, the present invention may be used internally by authorized employees of a retailer and/or its delivery service. It may also be designed to be accessible over the Internet by customers of an on-line retailer or the like. In the former embodiment, the user of the software is likely an authorized employee. In the latter embodiment, the user may be the actual customer, inputting information from a remote data sharing device such as a personal computer, personal telecommunications device (e.g., cell phone), palm top computer, etc.

In order to begin developing and optimizing a delivery schedule, the scheduling software must have certain initialization parameters. Such parameters can be pre-coded as defaults into the neural network, or they can be customized by the user. For example, the maximum number of delivery vehicles, the maximum number

of deliveries per vehicle, the maximum cost allowed for a delivery, and the form of the fuzzy cost determination function are preferably made available to the neural network before adding deliveries to a blank schedule. There are two possible sources of this information. The first is for the information to be generated internally by the software itself. The second option is to import this information from an external data file or interaction with the user.

The neural network (or genetic algorithm) should also be given general information about the customers to whom deliveries will be made. Some examples in the preferred embodiments are the distance and travel time from a given customer to every other customer and an estimate of the time needed at each location. Regarding the latter, either default or input data can be used. This allows the user to utilize historical or other means to determine the length of time needed on a stop-by-stop basis or to use a predefined default or algorithm if better information is not available. The information on time and distance is pre-calculated using geocode-type or other software in order to save time during the schedule optimization process. Provision is made for the software to utilize different time and distance matrices for different days and times within days. This makes adjustment for rush hour, holidays, construction, etc. easy and allows for much better predictions.

PARAMETERS

Figure 1 shows some of the parameters preferably used in the present invention, and in particular it shows a sample dialog box 5 that is displayed to the authorized user of the system. This example also contains parameters relevant to a simulation developed for that embodiment. The format and appearance of the dialog box need not be exactly as shown in Figure 1.

In the figure, the Max Values area 10 includes the maximum number of trucks or other delivery vehicles 10a that can be used in the schedule, the number of customers included in the simulation to be sampled from 10b, and the maximum number of legs 10c, i.e., the maximum number of deliveries/stops per vehicle per shift. Because each vehicle has a start time 30e and a stop time 30f, if there are going to be multiple shifts then the number of vehicles 10a may represent the number of physical vehicle/shift combinations that could exist. For example, if four trucks are used in two non-overlapping shifts, then parameter 10a may be set to eight. The vehicle capacity may be naturally limited by the time required to make deliveries or by the capacity of the vehicle to carry payload.

The number of customers 10b determines the dimension of a Customer ID Array that the simulation will produce, along with a Vehicle Distance Array and a Travel Time Array. In the real world application, these arrays are given to the program via an Application Program Interface (API).

Figure 1 also shows scheduling constants 20 that define the shape of the fuzzy cost determination function that is used in calculating the best delivery routes. The late arrival offset 20a (in units of time) sets the number of minutes late from "on time" that a predetermined proportion of the cost, e.g. 50%, should be applied in the cost evaluation of a late arrival.

An example of the fuzzy arrival time cost function, i.e. the fuzzy determination function, is given in Figure 2, which shows a combined effect of both late and early costs and helps illustrate some of the parameters of Figure 1. On its vertical axis is the cost for early (left side of horizontal axis) and late (right side of horizontal axis) arrivals.

The example of Figure 2 uses a late arrival offset set to 7.5 minutes, seen at point 201, with the late arrival cost 20c set to 120 and the early arrival cost 20h set to 20. The default early arrival cost is often set to zero (not shown in Figure 2).

The late arrival multiplier 20b (in units of time) is the number of minutes, centered on the late arrival offset 20a, between where a predetermined low proportion, e.g. 10%, and high proportion, e.g. 90%, of the late arrival cost 20c is applied in the cost function. In the example of Figure 1, the value 20b is set to 15 minutes, so in Figure 2 the multiplier 20b is a fifteen-minute wide space centered at point 201 between $(10\% * \$120 = \$12 \text{ cost})$ and $(90\% * 120 = \$108 \text{ cost})$. So zero minutes late incurred 10% and fifteen minutes late incurred 90% of the late arrival cost 20c.

The late arrival cost 20c (in monetary units) is a maximum amount of money, chosen as a fictional amount or estimated in terms of time and/or fuel, value of lost customers, and/or any other factor, that can be applied to the fuzzy cost function if a delivery is late. In the example of Figures 1 and 2, the value is set to \$120.

The late arrival line 20d (in monetary units per time) represents the dollars per hour to be added to the fuzzy cost function for each hour late. Figure 2 uses a late arrival line of zero, and thus it has a flat region 210. Figure 3 shows how the cost function increases at region 310 with a slope equal to Figure 1's late arrival line 20d of \$50 per hour.

The late arrival quad 20e (in monetary units per time) represents the dollars to be added to the fuzzy cost function for each hour squared late, essentially a quadratic term in the cost function. In the example of Figure 1, the value is set to 50. Translated into the cost function, Figure 4 shows how the increase in cost accderates for late arrivals at region 410. The arrival time parameters, in essence, set the window over which arrival is to be expected. The quadratic term is included to allow the user

to remove the possibility that a small number of stops will be scheduled very late. In some applications, sacrificing prompt arrival at a single or small number of stops can yield a schedule whose overall cost is lower. If this is undesirable, it can be eliminated.

5 Schedule constants 20 also include an early arrival offset 20f (in units of time) which is the number of minutes early from on-time that a predetermined proportion of the cost, e.g. 50%, should be applied in the cost evaluation of an early arrival. It is similar to the late arrival offset, and in the example of Figure 1 is set to 30 minutes, corresponding to point 202 in Figure 2.

10 The early arrival multiplier 20g (in units of time) is the number of minutes, centered on the early arrival offset 20f, between where a predetermined low proportion, e.g. 10%, and high proportion, e.g. 90%, of the early arrival cost 20h is applied in the fuzzy early arrival cost function. In the example of Figures 1-4, the value is set to 30 minutes.

15 The early arrival cost 20h (in monetary units) is the maximum value, created as a fictional number or estimated based on time and/or fuel and/or any other factor, that can be applied in the fuzzy cost function when a delivery is early. In the example of Figures 1-4, the value is set to \$20.

20 The overload cost 20i (in monetary units) represents the cost added to an evaluated schedule if the number of packages scheduled to be delivered by a vehicle exceeds the maximum value 30d set for that vehicle. This allows the vehicle's maximum capacity to be based on time or payload, whichever occurs first.

25 The cost threshold 20j (in monetary units) is used by the time slot generation routine to limit the cost of any delivery to a maximum value. If the incremental cost of adding a customer to the schedule at a given slot exceeds this value, the slot will

not be returned to the customer as being available. The value typically should be less than the late arrival cost 20c. In building schedules, this cost threshold value will affect the total schedule cost by restricting the cost of the slots presented to the customer. A higher value makes it possible for the customer to pick costlier slots, which, in turn, is likely to raise the expected cost for the final schedule. A low value will reduce the expected cost of the final schedule, but raises the possibility that a customer will not be offered a feasible slot due to cost considerations.

The default service time 20k sets the amount of time that a vehicle is expected to spend at each delivery stop. When manually adding customers to the schedule in a simulation, the service time can be set independently 802f (Figure 8) for each customer, and in real world applications it can be passed through the API or the default can be used.

The dialog box 5 also shows parameters related to the vehicles (or trucks) themselves. These vehicle parameters 30 include a set of parameters 30b through 30q that are preferably individualized for the vehicle designated by an identifier 30a. In simpler embodiments, all of the parameters 30b through 30q can apply to all of the vehicles, such as when the system uses a single type of vehicle. However, it is preferable to customize the parameters for each respective delivery vehicle. In Figure 1, the vehicle identifier 30a is numerical, but it can be the vehicle's license plate or any other means of identifying the vehicle.

The distance cost 30b (in monetary units per distance) is the cost in dollars per mile that should be associated with the specified vehicle. It can be based on fuel consumption, anticipated wear and tear, etc. The distance cost (and time cost discussed next) are often used in conjunction with the fuzzy arrival time function.

The time cost 30c (in monetary cost per time unit) is the cost in dollars per hour that should be associated with the specified vehicle and might be based on personnel wages, monthly depreciation, etc.

Bin capacity 30d is the total number of bins (or packages or whatever is being delivered, if any) that the vehicle can hold. If this capacity is exceeded in the schedule being evaluated, then the overload cost 20i is added to the cost of that schedule. This is included as an alternative to maximum legs for defining when a vehicle has reached delivery capacity. In some cases, the capacity of the truck will depend on the particular orders placed.

The start time 30e is the time of day that the vehicle departs on its route. Stop time 30f is the time of day that the vehicle should arrive back at the warehouse (or other location) at the end of its route. If the evaluated schedule has the truck returning to the warehouse late, then the cost for that schedule will include the late arrival cost 20c just as it is for a late arrival to a customer. Alternative embodiments can use a separate late-to-warehouse parameter, or exclude any late-to-warehouse cost.

The system preferably accommodates lunch-time and other breaks if the delivery schedule is long. Lunch time 30g is the duration (e.g., in minutes) of a lunch break; lunch start 30h is the earliest time of day that the vehicle's personnel can begin taking lunch; lunch stop 30i is the latest time of day that they can begin lunch. Break time #1 30j is the duration (e.g., in minutes) of a first break that the delivery personnel are permitted; break #1 start 30k is the earliest time of day that the vehicle's personnel can begin the first break begins; break #1 stop 30l is the latest time of day when they can start their first break. Break #2 time 30m, start 30n, and stop 30o are analogous to parameters 30j through 30l, but for a second break. The invention can accommodate additional breaks, or fewer, depending on the desires of the end user. The options

"use lunch in schedule" 30p and "use breaks in schedule" 30q determine whether the initialized schedule will contain lunches and breaks for the vehicles. These parameters are preferably examined only when the schedule is initialized, so if either one is changed, the schedule must be re-initialized for the change to take effect.

5 The dialog box 5 also shows Simulation Data 40, including the square size 40a in linear units such as miles, although it could accept square units. The square size 40a is the size of the geographic region that is serviced by the delivery system. This parameter is not used if customer information is imported from an external data file. The simulation included in the example embodiment can also artificially generate
10 customer data, and customer locations are randomly generated in this region. The authorized user can thus study delivery efficiency by defining a delivery region and a density of customers. Delivery cost is strong function of customer density in most applications.

 The average vehicle speed 40b (in linear units per time) is an estimated
15 average travel speed for the vehicle, to be used in generating customer data. The average speed is used with the distance information to determine the values for the travel time matrix. This parameter is not used if customer data is imported from and external data file.

 The Euclidean distance option 40c allows the system to determine the method
20 of calculating the distance between any two customers. When selected, straight line distances are used in the distance calculation. When not selected, city block distances, actual road lengths, or other types of distances are used. Alternatively, an externally generated customer matrix and associated data can be passed to the simulation, in which case, the time and distance data are also passed and are not

generated by the simulation. In real world applications, geocode software is generally used to calculate time and distance.

A random seed 40d is used to initiate a random number generator for building customer data. Using different values will place customers into different locations
5 along the delivery route. The random number generator is only used in the simulation for random selection of customer locations, choosing times, etc.

Simulation size 40e determines the number of customers that will be added to the schedule each time the build schedule command is executed in the simulation. If this value is set to 10 for instance, every time the operator tells the simulation to build the
10 schedule, the simulation will attempt to add 10 more fictitious customers to the then current schedule. This process can be performed manually or automatically in conjunction with the AutoBuild function. If, for a given customer, no available delivery time slots are generated, then that customer will not be added to the delivery schedule. The simulation size parameter 40e can be set to a small value when used in
15 conjunction with a nonzero value for the AutoBuild seconds parameter 40h to slowly add customers while concurrently running schedule optimization routine.

The user may select or deselect a random segment option 40f which, when set during the automated construction of a schedule, instructs the schedule builder to randomly select a segment during the day and randomly select a slot returned by the slot
20 generation function. When this parameter is not selected, slots are generated for all segments of the day and the lowest cost slot is selected by the build function. This allows the operator to study a theoretical lowest cost schedule as opposed to a more "real world" schedule.

The user may also select a random customer option 40g which, when selected
25 during the automated construction of a schedule, instructs the schedule builder to

randomly select one of the available customers for inclusion in the schedule. When this parameter is not selected, customers are selected sequentially from the list of customer identifiers found in the Customer ID Array.

AutoBuild parameter 40h (in time units) determines the time in seconds
5 between automatic calls to the build schedule function. Calls will be made continually at this time interval until this parameter is reset to zero or the AutoBuild iteration limit has been reached. This gives the user a way to test optimization of the schedule (which can be run concurrently) as customers are being added to the schedule. When using this parameter it is preferable to set its value to 15 seconds or
10 more in order to allow the schedule optimization routine time to improve the schedule. Too small a value will decrease the effects of optimization on the final schedule.

An AutoBuild iteration parameter 40i designates the maximum number of times that automatic calls to the build function will occur, thus allowing for a totally
15 automatic optimized simulation.

For the user's benefit, he or she can ask for information to be displayed in graphical 50a or text only 50b format, and with or without actual times 50c. Selecting the graphics format 50a generates the graphical representation of the schedule as shown in Figure 5. Although other formats are certainly possible. The originally generated
20 schedule 510 is displayed at the left side of the graphic, with a current optimized schedule 512 displayed at the right. If the delivery system uses more than one vehicle, then it is preferable to use different colors or shadings to designate the routes of different vehicles. By placing the original and updated schedules adjacent one another, the user can make a visual comparison in order to see how it has changed due
25 to the optimization. The original schedule 510 is updated when a build schedule

command is entered. This allows the user to view the effects of adding customers on the latest schedule produce by the simulation.

5 When the Text Only option 50b is selected, the display becomes a text representation of the current optimized schedule. An example is shown in Figure 6. The times shown represent the scheduled arrival times for the customer.

 The statistics shown in the graphics display of Figure 5 are shown in Figure 6 as "customer identifier/scheduled delivery time slot," with each column
10 corresponding to a respective vehicle. The customer identifier is, for example, a three digit number and the target arrival time is expressed in 24-hour time, hh:mm. Starting times for lunch and breaks are denoted by LN and BR, respectively. Also shown are the average distance between the warehouse and all of the customers, the average distance per leg in the schedule and the actual number of legs in the schedule, all in
15 region 602.

 The display preferably uses different shades or colors to designate the type of delivery order. Normal customer orders are shown in black, for example, while available slots are color coded, e.g. in blue. Available slots will either be converted to real orders or removed at some point in time before final schedule optimization is
20 performed. Ghost orders (orders that are used to force the software to start vehicles from specified regions, e.g. outlying areas) may also be used as the available slots, and should be removed from the schedule before final schedule optimization is performed. A command exists to automatically perform this removal.

 When the Actual Times option 50c is selected, a text representation of the current
25 optimized schedule is displayed. An example is shown in Figure 7. The times shown

represent the actual arrival times for the customer. Actual arrival times are the times the program believes the truck will actually arrive at the destination based on the time matrix. These times may be different from the target arrival times. Based on the fuzzy cost determination function, the program allows the optimizer to rearrange the schedule so that expected arrival times may be different from targets if the cost is still below the maximum allowable cost and the overall cost for the schedule is reduced.

Additional parameters relevant to the neural network or genetic algorithm can preferably be changed by an authorized user. Three such parameters, Beta, Gamode, and NConsensus are described below.

Beta is a parameter which controls the probability that higher cost schedules are accepted. The higher the value, the less likely a solution will be accepted into the genetic algorithm popthat has a higher cost than the best schedule.

Gamode is a parameter that controls whether the parent or the worst solution is replaced during the optimization. The parameter will therefore have one of two possible values, and simply acts as an identifier of which of the two to replace. Any two values can thus be used; other unrecognized values may cause the genetic algorithm to malfunction.

NConsenses is the number of schedules in the population being used by the genetic algorithm. A default value can exist, such as 8. A larger value will increase the time it takes to find an "optimal" solution. Too small a value will give poorer solutions. This parameter should only be changed if the schedule optimization is stopped and any change should be followed by a schedule initialization command.

SLOT GENERATION AND SCHEDULE OPTIMIZATION

The slot time generation routine analyzes the cost related to placing that customer in the delivery schedule using the fuzzy cost determination function at time slots that are spaced by predetermined time spacings, such as every 5 minutes (although any spacing could be selected). If the time segment is input, then this cost analysis is preferably done only for the time segment of the day that the user specified. For systems that control more than one delivery vehicle, this cost analysis routine is also preferably done for more than one vehicle, and preferably for every vehicle in the schedule. However, the artisan will appreciate that only some of the vehicles may be suited for delivering a particular item to that customer, such as when the item is very large and requires a larger vehicle, or if it is a frozen item and requires a refrigerated vehicle, etc. Regardless, the software routine analyzes the cost based on one or more of parameters 802b-802f, and the resulting cost increase(s) are placed in a sorted list. The top few, e.g., four, are returned to the calling process.

Distances between previous delivery locations and the new customer's address, and the streets that each vehicle would take in order to travel from one destination to the next, are evaluated by using geocodes or other known systems, as explained in more detail below. In the end, the slot generation routine sorts the available slots according to cost and, in the embodiment reflected in Figure 8, returns the four best slots to the user's display, at region 804. Behind the scenes, the slot generation routine stores a data record identifying the four slots, the customer's identifier (a valid identifier from the customer ID array), and the other pertinent data for that delivery.

The customer or user decides which one of the four time slots is most convenient, and selects it using a mouse, keyboard, etc. This time slot is loaded into the customer ID array under that customer's identifier and, from there, is retrieved by

the optimization routine in order to include that customer into the schedule. The optimization routine may also load the customer ID array with the identity of the vehicle and leg on which that customer will be serviced.

In the event that the slot generation routine encounters a tie in ranking of
5 available time slots, the software can choose a winner based on any criteria. In one embodiment, an added cost related to the closeness in time to currently selected times is used to break the tie. For example, in the time period from 8:00 to 12:00, the added cost for a given delivery might be \$5. If all 5-minute slots have the same cost, then the first slot selected is 8:00. The closeness cost is then added to the remaining slots
10 on the list. This cost is calculated, in one example, as a price such as \$50, divided by (one plus the square of the scaled difference in time, e.g. in minutes, between the selected slot and the slot under consideration). By using this method the next slot generated would be 12:00 since it is farthest from the first selection. Then using 8:00 and 12:00 as selected slots, the third selection would be 10:00, midway between the
15 first two, using this closeness cost criteria. In simpler embodiments, the software can settle a tie simply by choosing the first few slots immediately preceding or following a slot that is already on the schedule. This method allows the slot generation routine to space possible times suggested to the customer across a segment of the day.

The time slot generation function preferably does not report a slot as being
20 available if the incremental cost for that slot exceeds the cost threshold 20k. In building schedules, this cost threshold setting 22k will affect the total schedule cost by restricting the cost of the slots presented to the customer. A higher threshold value 22k will make it possible for the customer to pick costlier slots. The effect will be to raise the expected cost for the schedule, though additional stops may be added. A low
25 value will lower the expected cost of the schedule while raising the possibility that a

customer will not be given a feasible slot based on cost considerations. Experimenting with different threshold values 22k will aid in determining an optimal value for the environment in which the invention is to be used.

The present invention preferably employs geocodes, or latitudinal and
5 longitudinal coordinates, expressed in degrees, hours, minutes, seconds, and tenths-of-seconds of arc. Software databases currently exist which are able to translate street addresses into such coordinates, such as Navtech and GDT, so they need not be explained in further detail herein. These databases are able to generate physical distances and time-of-travel between street addresses which the slot generation
10 routine uses in order to evaluate the cost of adding a new customer to a particular slot within a schedule. The artisan will appreciate that certain embodiments of the present invention can omit physical distance and/or time-of-travel as elements of cost. Nevertheless, these pieces of data can be extremely useful in an evaluation of cost.

Schedule Optimization

15 In a preferred embodiment of the present invention, the schedule optimization routine is a server that operates continually, improving the cost efficiency of the delivery schedules. In alternative embodiments, and in particular low-volume applications, it can be called only when a new customer is added.

In the example being described, whenever new customers' delivery time slots
20 are obtained by the time slot generation routine, the server tries to improve the cost effectiveness of the overall delivery schedule for all vehicles while retaining, as best it can, the customers' requested delivery time slots. In certain situations, a customer who requested a particular time slot will have his or her time slot adjusted. However, the present invention places constraints on how much the scheduled time slot can
25 differ from the requested time slot. In its most basic sense, the optimization routine

re-allocates the deliveries by switching some customers to different vehicles, by moving the legs of certain vehicles' routes, or by changing the actual delivery times for certain customers so that, in aggregate, the cost for the entire delivery schedule for all vehicles is minimized.

5 The schedule optimization is best performed using a genetic algorithm. This algorithm builds a population of solutions and attempts to improve a cost function by making changes to the schedule. A genetic algorithm modifies a schedule using two basic methods. The first is a mutation operation. In the case of a schedule, the mutation operation is simply to move a leg from its original position in the schedule
10 to a new position in the schedule. This could mean changing either the vehicle and/or the delivery sequence for this leg. The second operation is called a crossover operation. This term comes from the genetic analogy of reproduction where two different genes are cut at a random point and the tails are switched to form two new genes. For schedules or lists, crossover operations include switching two vehicles'
15 schedules at a certain point in the day, by reversing the order of delivery for a certain sequence of legs, or by switching the positions of two legs in the schedule. Any of these changes will cause the Distance Array, Time Array, and/or Customer ID Array to be changed. These arrays are described below.

Each modified schedule is evaluated based on the cost function. If the cost
20 improves, then the improved schedule is added back into the population of solutions. Although the number of possible solutions (population) can simply be increased, that would also increase future processing time. So it is preferable to have a limit on the number of solutions to be evaluated, such as eight although any number can be used, and to have the new solution replace either the parent (previous version of that
25 solution) or the worst (most costly) schedule in the population.

If the cost does not improve, the newly created schedule may still end up in the population. In one embodiment, the difference in cost between the then-generated solution and the best (least costly) solution is used to generate a probability of acceptance (e.g., a percentage). The greater the difference the lower the probability of acceptance. A random number (e.g., between zero and one) is then generated. If the random number is lower than the probability of acceptance then the schedule is added back into the population. Otherwise it is discarded.

The process of generating and evaluating solutions is preferably repeated continually during the optimization. The best schedule found during this process is saved for use by the slot time generation algorithm and for actual deliveries. That is, time slots that are free in the best schedule are made available to the slot time generation routine for selection and (optional) presentation to the user. See Figure 8. This is done by setting a flag for all available time slots in the schedule array for that particular vehicle. When departure time for one or more of the vehicles arrives, the optimization routine excludes that vehicle's schedule from those that are manipulated. The fixed schedule is printed and the vehicle's driver is given the list of customers and their addresses, preferably also with the geocode software's street directions on how to travel from one destination to the next.

Schedule Cost Function

One of the keys to the optimization routine is the schedule cost evaluation function. The neural network uses this function to evaluate schedules. The specification of the cost function is best when it closely represents realistic costs encountered in a schedule. It is clear that distance related costs should be part of this cost function. This cost can be implemented in various manners, one of which is to specify a cost per mile traveled (or other linear measurement) and/or a cost per minute

of travel time (or other unit of time) for each of the vehicles in the schedule. These costs are specified in the vehicle structure as Distance Cost and Time Cost. The vehicle structure is essentially a data record specific to a vehicle, storing data relevant to that vehicle's operation in the system. Some or all of the information displayed in sections 20 and 30 of dialog box 5 can be part of the vehicle structure. Optionally, additional information such as an identification of the personnel who man the vehicle, the vehicle's government registration information, and the like can also be stored in the vehicle structure. The vehicle structure is preferably accessible by both the slot time generation routine and the optimization server.

10 If distance related costs were the only costs in the schedule, the neural network or genetic algorithm optimization would not care if a customer received their order late or early. However, the cost of being late by a certain time can be detrimental as well. If late means greater than 15 minutes, it is undesirable to have the cost of being 14 minutes late be zero, and 15 minutes late to be \$50 (or whatever cost is
15 designated).

Because constraints are difficult if not impossible to program into the genetic algorithm, they are best implemented as part of the cost function. For example, each truck should have a lunch or dinner break. If the genetic algorithm assigns two lunch breaks to the same truck (e.g. through a mutation that moves lunch from vehicle #j to
20 vehicle #k), then the cost evaluation should tell the algorithm that this is infeasible by returning a high cost for that schedule. For example, the system could set an enormous constraint cost, such as \$999999.00, to insure that no such constraints would be violated.

Controlling the Optimization

There is no need to run optimization until at least three legs from customers are in the schedule. As more legs are added to the schedule, the optimization becomes more effective. The implementation of the present invention preferably follows some basic characteristics or rules related to optimization. First, the larger the schedule (number of legs), the harder it is to find an optimal solution. The difficulty grows by $(n!)$ or "n factorial," where "n" is the number of legs in the schedule. Second, the more constrained the schedule, the less opportunity there is for improvement. This can be seen by increasing the number of vehicles for an optimized schedule. With a larger number of vehicles, there are a larger number of ways that a schedule can be implemented feasibly. Also, as more constraints are added, such as early arrival costs, the cost of the "optimal" schedule increases. Third, it is very unlikely that the best solution found for a schedule is actually the "optimal" schedule, mathematically speaking. Because of the large number of possible solutions, $(n!)$, only a small fraction will be investigated by the neural network. Once the optimization routine reaches its best solution, the changes to the schedule that would improve it are unlikely to be found by the genetic algorithm through the random methods used. When late arrival costs are set to zero and only distance costs are optimized, the visual display of the best schedule appears very close to optimal. For instance, a vehicle crossing its own path is a sign of a less than optimal solution. Clearly, the optimization routine described herein could be improved by adding or changing constraints, but the processing cost that comes with such modifications will likely not improve performance in a realistic sense.

As described herein, the optimization is handled in a separate thread from the slot time generation. The system is designed so that both operations can occur

concurrently. Thus the best schedule can be changed as slot times are generated. A different embodiment contemplates running the optimization routine only after the slot time generation. This reduces processing requirements.

It should also be noted that the structures outlined above makes it possible to
5 run several schedule programs independently and then recombine them into a single optimization later. This allows, for example, that vehicles can be removed from the overall schedule and then reinserted. It also allows for independently scheduling a remote region and then putting that schedule into the master schedule for optimization. If, for instance, the user wishes to operate deliveries in a remote region,
10 is uncertain of the number of deliveries which will be requested and wishes to fill all requested. He can operate an independent schedule until a cutoff time, for example, and then insert the truck into the master schedule. The optimizer may be able to schedule other deliveries on the way to or from the remote region. As is the case with using Ghosts, the region must be truly remote to require this approach, as the normal
15 program operation is highly efficient in scheduling most regions, even those that are not uniform in the density of delivery locations.

Correcting Ghost Times

As indicated previously, the slot time generation routine in rare instances can
20 have trouble finding slots for customers in outlying areas due to the remoteness (usually meaning separated by large distances across areas with no delivery stops) from the warehouse or other starting point. By using ghost orders, a vehicle can be forced into an outlying area as previously described.

Ghosts are added to a schedule by selecting and adding fictional customers
25 from the area in a normal fashion. Optionally, a minimum distance solution can be

calculated for the selected customers by setting late arrival cost to zero and running the optimization routine on those fictional customers, or ghosts. This produces a minimum distance solution but contains bad arrival times for the ghosts. A routine exists to change the target arrival times for the ghosts and distribute them evenly in
5 time across the shift. This prevents a situation in which all ghosts are bunched together at the beginning of the vehicle's route.

Thereafter, the user can remove one or more of the ghosts periodically, so that they are slowly phased out of the schedule. Alternatively, the optimization routine can remove a ghost (as recognized by its Type) whenever a new customer is added to
10 that schedule near the Ghost reversion. Such replacement can occur each time a new customer is added, or every other time, or on some other periodic basis. Before the schedule is finalized, however, the optimization routine should scour the customer Types and remove any remaining ghosts.

Additional Variations

15 What results at the end of processing is a set of routes for the delivery vehicles to take. The routes will bring the vehicles to their appointed destinations in a most efficient manner, with only minimal likelihood that deliveries will be later than the window originally defined by the cost arrival function.

The invention having been thus described, it will be obvious that the same
20 may be varied in many ways not only in construction but also in application. For example, the parameters described herein are expressed in terms of monetary or other "real" units. Such need not be the case; the parameters can be designated in fictional units. However, using real units permits the cost estimations to be used in delivery charges to the customer. A customer can be shown the approximate cost for a
25 selected time slot, and only upon agreeing to pay that cost will the time slot be

forwarded to the optimization routine. The cost function can also be used on the actual delivery schedule that took place. The vehicle's log can be input and evaluated against the optimized schedule in order to see whether and how much money should be credited to the customers' accounts due to the fact that a delivery was late.

- 5 In addition, although the present description speaks of "minimizing" costs, analogous parameters can be set up in a fashion that makes the slot generation routine and the optimization thread try to maximize the variables in the schedule. Such variations are not to be regarded as a departure from the spirit and scope of the invention, but rather as modifications intended to be encompassed within the scope of the following claims.

10

CLAIMS

What is claimed is:

1. A method of scheduling visits by at least one vehicle to a plurality of
5 location-times, the method comprising the steps of:
collecting the plurality of location-times (510, 512);
defining a plurality of possible delivery schemes (510, 512), with each
delivery scheme providing for each location-time to be visited by a vehicle of the at
least one vehicle, and with each delivery scheme including projected actual times for
10 each visit;
for each delivery scheme determined in the defining step, calculating an
efficacy value for the delivery scheme; and
choosing a delivery scheme from the plurality of delivery schemes with the
greatest efficacy value (512) as the chosen delivery scheme.
15
2. The method of claim 1 further comprising the step of:
dispatching the vehicles of the at least one vehicle to follow the chosen
delivery scheme.
- 20 3. The method of claim 1 wherein there is a fuzzy correlation between
calculated efficacy value and the variance between time preferences of the location-
times and the projected actual times, so that efficacy cost increases substantially
smoothly as the variance increases.

4. The method of claim 3 wherein efficacy cost increases at a greater rate as the variance increases.

5. The method of claim 1 wherein the calculation efficacy value is based on at least one of the following considerations: distance traveled by the at least one vehicle under the delivery scheme; amount of fuel used by the at least one vehicle under the delivery scheme; loading of cargo under the delivery scheme; and time constraints.

10 6. The method of claim 1, wherein the determination step is based at least one of the following: projected vehicle speeds, projected traffic conditions, actual vehicle travel speeds and actual travel conditions.

7. The method of claim 1 wherein, in the calculating step, a predetermined condition causes any delivery scheme incorporating the predetermined condition to have an enormous efficacy cost, whereby any delivery scheme incorporating the predetermined condition will be extremely unlikely to be chosen as the chosen delivery scheme at the choosing step.

20 8. The method of claim 7 wherein the predetermined condition is the condition of a vehicle having too many breaks.

9. The method of step 1 wherein the collecting step, the defining step, the calculating step and the choosing step are repeatedly performed as the vehicles of the at least one vehicle are visiting the plurality of location-times.

10. The method of step 1 wherein the collecting step is performed using a computer network.

5 11. The method of step 1 wherein at least some of the location-times are ghost location-times, which represent vehicle visits for the purposes of scheduling, but which are not intended to cause or reflect actual vehicle visits.

12. The method of step 11 further comprising the following steps:
10 communicating the scheduling to the respective operator(s) of the at least one vehicle; and
prior to the communicating step, removing any ghost location-times from the scheduling.

15 13. The method of claim 11 wherein the ghost location-times are added to emulate actual location-times that are expected to be added to the schedule.

14. The method of claim 13 wherein the ghost location-times are based at least partially on historical experience.

20

15. A method of scheduling visits by at least one vehicle to a plurality predetermined locations, the method comprising the steps of:
collecting a plurality of predetermined location-times (510, 512);
collecting a first proposed additional location-time (801);

- defining a plurality of possible delivery schemes (510, 512), with each delivery scheme providing for each predetermined and proposed additional location-time to be visited by a vehicle of the at least one vehicle, and with each delivery scheme including projected actual times for each visit;
- 5 for each delivery scheme determined in the defining step, calculating an efficacy value for the delivery scheme;
- choosing a delivery scheme from the plurality of delivery schemes with the greatest efficacy value as the first chosen delivery scheme (512); and
- deciding whether to accept or reject the first proposed additional location-time
- 10 as a predetermined location-time for an actual visit based at least in part on the efficacy value of the first chosen scheme.

16. The method of claim 15 further comprising the steps of:
- collecting a second proposed additional location-time, along with any
- 15 applicable cargo that may be transferred to or from the at least one vehicle at the second proposed additional location-time; and

 performing, in the case that the first proposed additional location-time is rejected at the deciding step, the following further steps:

- re-defining a plurality of possible delivery schemes, wherein each
- 20 delivery scheme allows each predetermined and proposed additional location-time to be visited by at least one vehicle;

 for each delivery scheme determined in the defining step, re-calculating an efficacy value for the delivery scheme, wherein the efficiency value reflects the overall efficacy of the delivery scheme;

choosing a delivery scheme from the plurality of delivery schemes
with the greatest efficacy as the second chosen delivery scheme; and
deciding whether to accept or reject the second proposed additional
location-time as a predetermined location-time for an actual visit based at least
5 in part on the efficacy value of the second chosen scheme.

17. The method of claim 15 wherein the first location-time and the second
location-times specify identical locations and cargo, but have different associated time
preferences.

10

18. A method of setting an appointment from a plurality of proposed
appointments, the method comprising the following steps:
collecting a plurality of predetermined appointments (510, 512);
defining a plurality of possible scheduling schemes (510, 512), wherein each
15 scheduling scheme allows each predetermined appointment to be accommodated,
while also allowing at least one proposed appointment to be accommodated;
for each scheduling scheme determined in the defining step, calculating an
efficacy value for the scheduling scheme; and
choosing a scheduling scheme from the plurality of scheduling schemes (512)
20 with the greatest efficacy value as the scheduling delivery scheme.

19. The method of claim 18 wherein the appointments represent visits by
delivery vehicles to location-times.

25 20. The method of claim 18 wherein:

the predetermined appointments represent firmly-scheduled visits by vehicles to make deliveries at associated location-times; and

the proposed appointments represent alternative delivery location-times proposed by a customer.

5

21. A method for scheduling travel of one or more vehicles, comprising:

(a) selecting a time slot from among a plurality of time slots, the selected time slot designating a time of vehicle arrival; and

(b) incorporating the selected time slot into an existing travel schedule for
10 one or more vehicles.

22. The method of claim 21, wherein the existing travel schedule includes a plurality of scheduled time slots, and further wherein said incorporating step includes:

15 (c) generating a new travel schedule, the new travel schedule including all scheduled time slots and the selected time slot;

(d) evaluating a cost of the new travel schedule versus a cost of the existing travel schedule; and

(e) replacing the existing travel schedule with the new travel schedule
20 based upon said evaluating step.

23. The method of claim 22, wherein the existing travel schedule includes respective travel routes for a plurality of vehicles, each travel route including a plurality of the scheduled time slots.

25

24. The method of claim 23, wherein said generating step moves at least one scheduled time slot from a first vehicle's travel route in the existing schedule to a second vehicle's travel route in the new travel schedule.

5 25. The method of claim 22, further comprising:
repeating steps (c), (d), and (e) a plurality of times for each execution of said step (b).

26. The method of claim 21, further comprising:
10 repeating steps (a) and (b) for a new selected time slot, the new selected time slot being different than the selected time slot.

27. The method of claim 21, wherein the existing travel schedule includes respective travel routes for a plurality of vehicles, the travel route for a respective
15 vehicle including a plurality of scheduled time slots, each time slot designating a time when the vehicle is to arrive at a destination.

28. The method of claim 21, wherein the existing travel schedule includes a plurality of scheduled time slots, and further wherein the plurality of time slots used
20 in said selecting step are different from the scheduled time slots.

29. The method of claim 21, wherein said selecting step includes:
evaluating respective cost increases to the existing travel schedule for a plurality of the time slots;

offering, to a user, a plurality of best time slots based on said evaluating step;
and
inputting, by the user, a choice of one of the best time slots, wherein the one
chosen time slot is the selected time slot.

5

30. A method of scheduling appointments, the method comprising the
steps of:

collecting the plurality of appointments (510, 512);

defining a plurality of possible schedules (510, 512), with each appointment
10 being accommodated in each possible schedule, and with each schedule including
projected actual times for each appointment;

for each possible schedule determined in the defining step, calculating an
efficacy value for the possible schedule; and

choosing a chosen schedule from the plurality of possible schedules (512) with
15 the greatest efficacy value as the chosen schedule.

31. A method of scheduling appointments, the method comprising the
steps of:

collecting a plurality of predetermined appointments (510, 512);

20 collecting a first proposed additional appointment (801);

defining a plurality of possible schedules (510, 512), with each possible
schedule providing for each predetermined and proposed additional appointment to be
accommodated, and with each possible schedule including projected actual times for
each appointment;

for each possible schedule determined in the defining step, calculating an efficacy value for the schedule;

choosing a schedule from the plurality of possible schedules (512) with the greatest efficacy value as the chosen schedule; and

- 5 deciding whether to accept or reject the first proposed additional appointment as a predetermined appointment for an actual appointment based at least in part on the efficacy value of the chosen schedule.

10

1/8

SIMULATION PARAMETERS		X	
MAX VALUES		SIMULATED DATA	
TRUCKS	1 — 10a	SQUARE SIZE (ML)	25 — 40a
CUSTOMERS	30C — 10b	AVG. SPEED (MPH)	30 — 40b
LEGS	30 — 10c	<input checked="" type="checkbox"/> USE EUROPEAN DISTANCE	40c
SCHEDULE CONTENTS		RANDOM SPEED	777 — 40d
LATE ARRIVAL OFFSET	75	SIMULATION SIZE	500 — 40e
LATE ARRIVAL MINS.	15	<input checked="" type="checkbox"/> RANDOM SEGMENT SELECT	40f
LATE ARRIVAL COST (\$)	120	<input checked="" type="checkbox"/> RANDOM CUSTOMER SELECT	40g
LATE ARRIVAL LINE	0	AUTOBUILD SECONDS	0 — 40h
LATE ARRIVAL QUAD (\$/HR)	0	AUTOBUILD	0 — 40i
EARLY ARRIVAL OFFSET	30	GHOST PERCENTAGES	100 — 40j
EARLY ARRIVAL MINS.	30	DISPLAY	
EARLY ARRIVAL COST	20	<input type="checkbox"/> GRAPHICS	50a
OVERLOAD COST	1	<input checked="" type="checkbox"/> TEXT ONLY	50b
COST THRESHOLD	50	<input type="checkbox"/> ACTUAL TIMES	50c
DEFAULT SERVICE TIME	10		
TRUCK PARAMETERS		OK	
TRUCK NO.	0	CANCEL	
DISTANCE COST	0 35 — 30b		
TIME COST	0 72 — 30c		
BIN CAPACITY	80 — 30d		
START TIME	08:00 — 30e		
STOP TIME	16:00 — 30f		
LUNCH TIME	00:30 — 30g		
LUNCH START	11:00 — 30h		
LUNCH STOP	13:00 — 30i		
BREAK 1 TIME	00:15 — 30j		
BREAK 1 START	9:00 — 30k		
BREAK 1 STOP	10:00 — 30l		
BREAK 2 TIME	00:15 — 30m		
BREAK 2 START	14:00 — 30n		
BREAK 2 STOP	15:00 — 30o		
<input checked="" type="checkbox"/> USE LUNCH IN SCHEDULE	30p		
<input checked="" type="checkbox"/> USE BREAKS IN SCHEDULE	30q		

FIG. 1

2/8

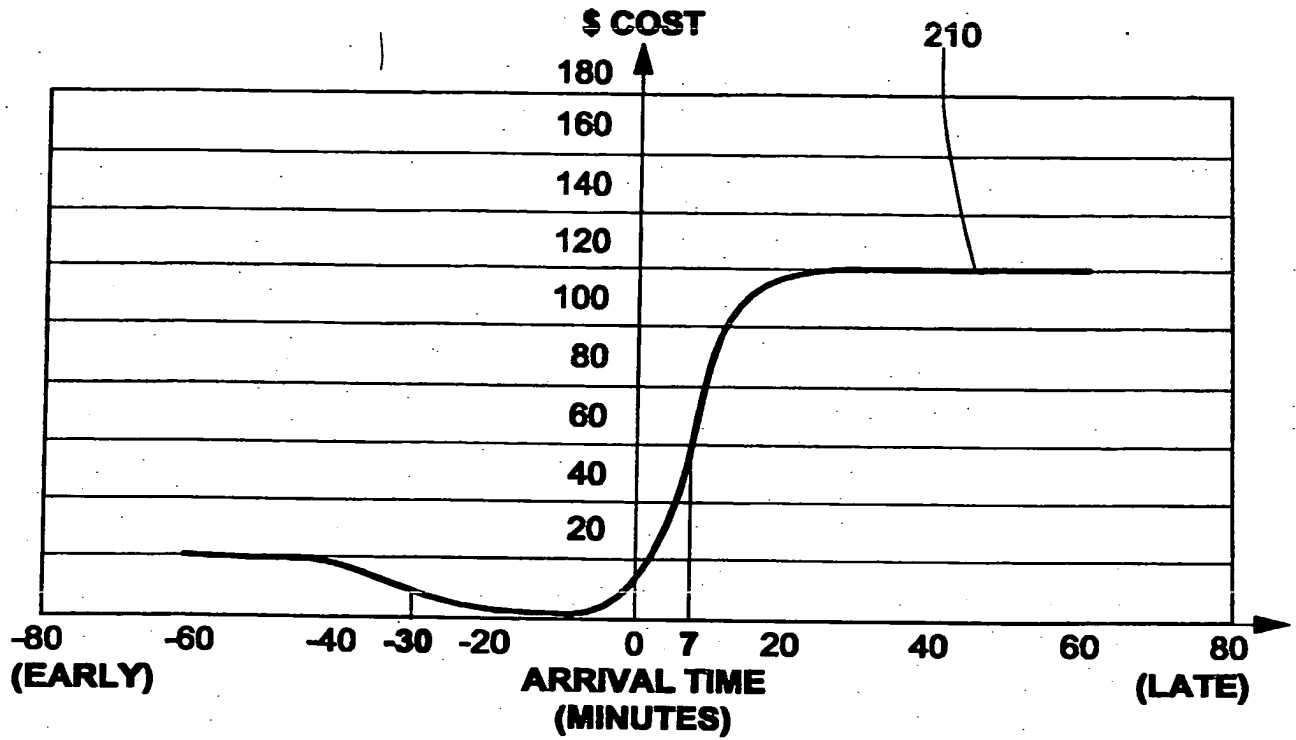


FIG. 2

3/8

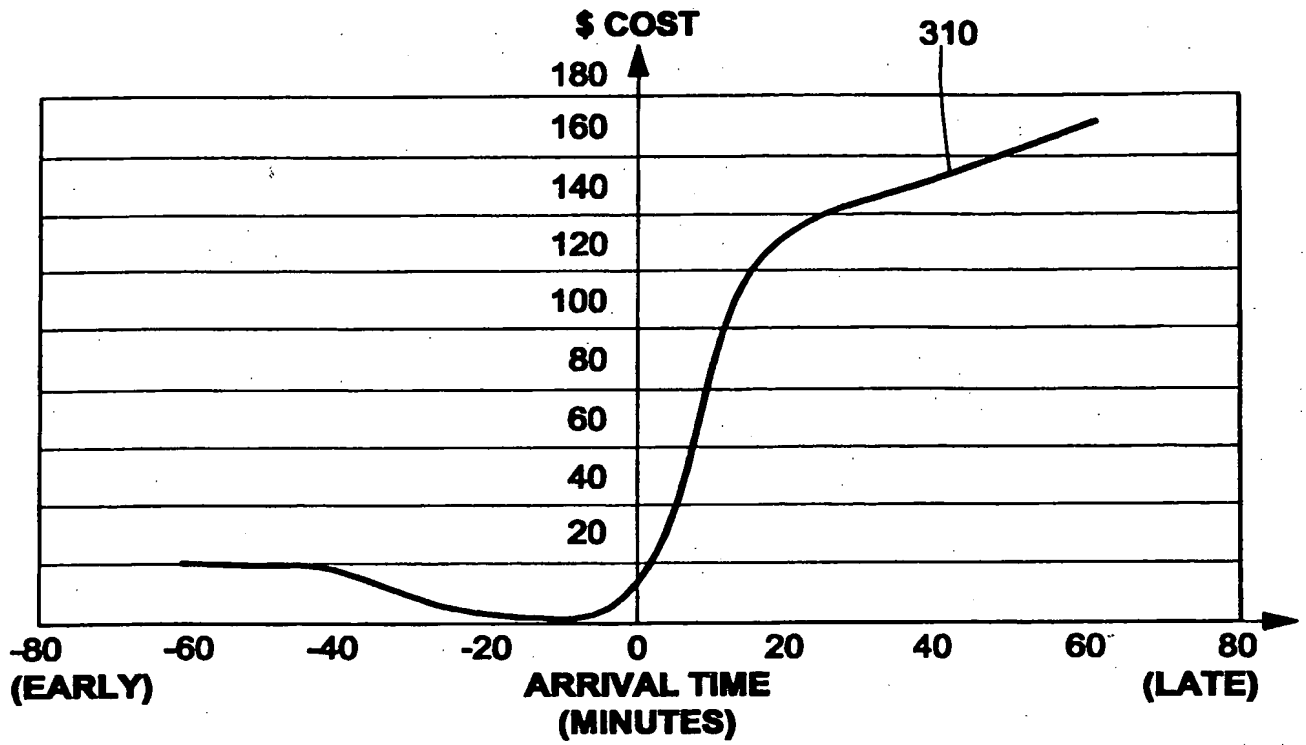


FIG. 3

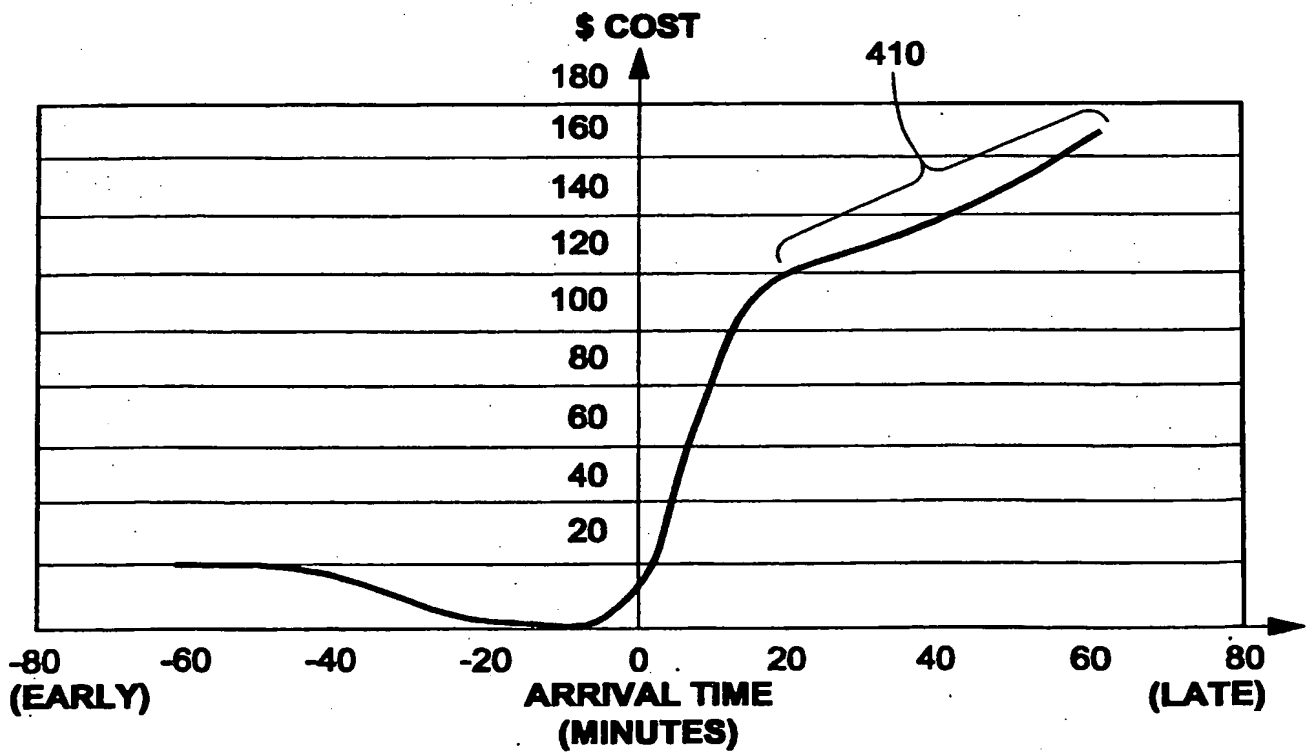


FIG. 4

4/8

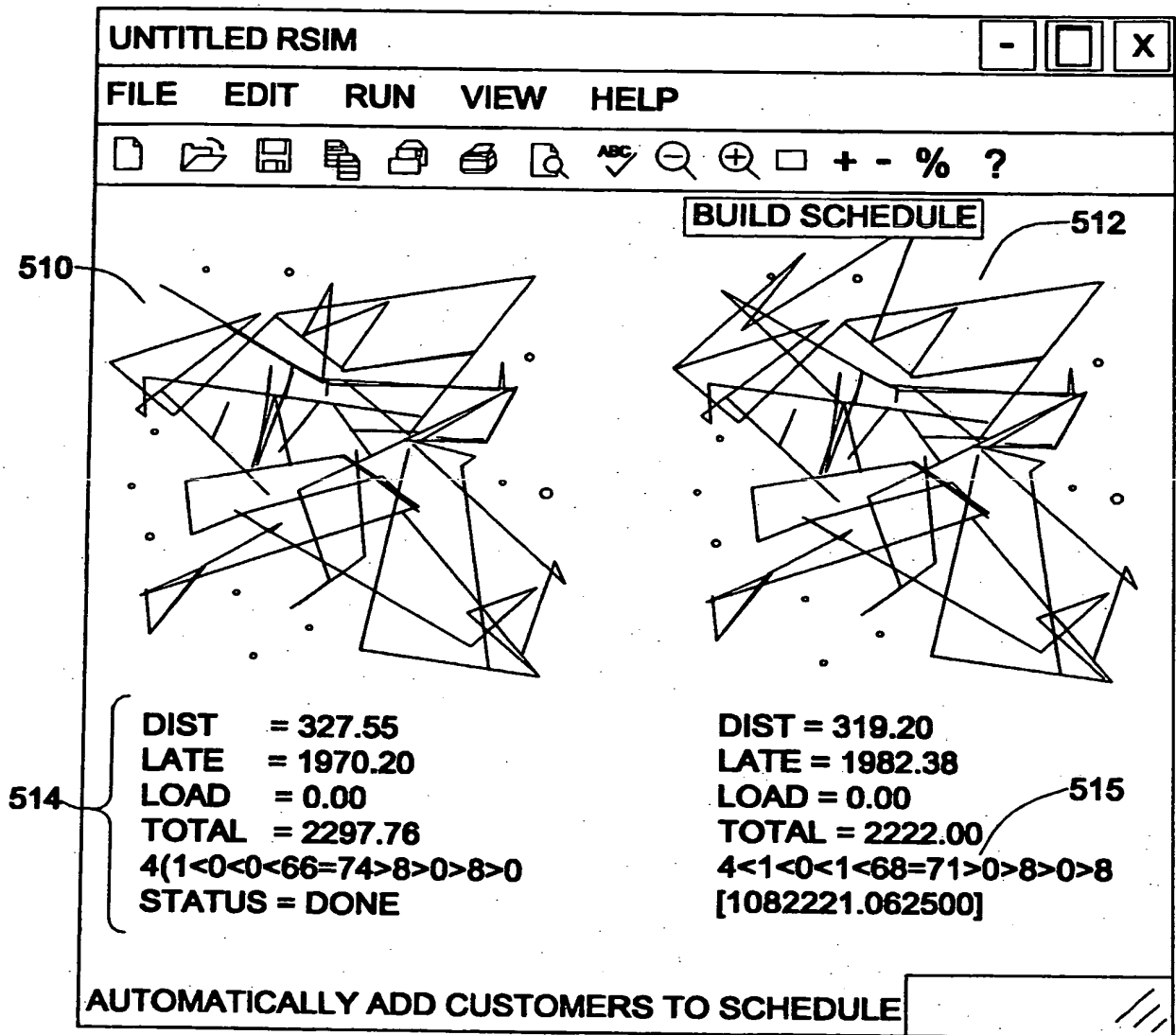


FIG. 5

5/8

UNTITLED RSIM															- [X]	
FILE EDIT RUN VIEW HELP																
[Icons] [?] + - %																
0>00:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>16:00	0>16:00	0>16:00
472>00:15	379>16:36	313>08:25	231>16:20	341>08:15	529>16:10	375>08:15	338>16:10	224>08:30	117>16:25							
423>00:35	226>16:50	462>08:50	112>16:46	352>08:30	487>16:25	512>08:30	243>16:36	156>09:10	109>16:50							
533>00:55	BR1>17:00	BR1>09:01	483>16:55	194>09:00	587>16:45	509>08:55	164>16:55	134>09:35	202>17:20							
566>09:10	311>18:00	596>09:25	382>17:20	364>09:30	484>17:05	373>09:10	319>17:25	BR1>09:47	BR1>17:31							
137>09:35	377>18:15	146>10:00	BR1>17:40	BR1>09:40	553>17:30	BR1>09:21	BR1>17:35	127>10:10	468>17:55							
BR1>09:40	300>18:40	291>10:25	296>18:00	400>10:15	BR1>17:41	100>09:41	556>18:00	439>10:25	402>18:20							
595>10:05	250>19:00	457>10:40	253>18:15	363>10:25	479>18:00	467>10:30	295>18:30	131>11:00	592>18:55							
525>10:30	519>19:15	176>11:00	430>18:25	520>10:46	297>18:40	167>11:05	163>18:50	374>11:45	548>19:20							
390>10:55	109>19:40	LN>11:13	372>18:55	554>11:00	213>18:56	354>11:15	259>19:15	511>12:10	376>19:45							
394>11:35	190>20:00	384>11:55	143>19:25	454>11:37	443>19:10	248>11:50	171>19:55	217>12:48	337>20:00							
300>11:50	LN>20:50	572>12:06	335>20:00	410>12:00	475>19:30	536>12:00	LN>20:10	LN>12:58	LN>20:15							
446>12:00	BR2>22:00	516>12:36	LN>20:10	210>12:20	168>19:45	252>12:30	BR2>22:00	232>13:29	BR2>22:00							
LN>12:12	0>24:00	501>13:00	BR2>22:00	148>12:40	220>20:00	409>12:50	0>24:00	161>13:45	0>24:00							
425>13:10	0>00:00	547>13:20	0>24:00	LN>12:50	LN>20:10	LN>13:00	0>00:00	296>14:15	0>00:00							
104>13:20	0>00:00	263>13:36	0>00:00	153>13:30	BR2>22:00	106>13:30	0>00:00	BR2>14:27	0>00:00							
422>13:55	0>00:00	444>14:25	0>00:00	542>13:55	0>24:00	346>13:55	0>00:00	258>15:00	0>00:00							
158>14:10	0>00:00	BR2>14:35	0>00:00	428>14:15	0>00:00	236>14:05	0>00:00	239>15:25	0>00:00							
BR2>14:22	0>00:00	392>15:05	0>00:00	BR2>14:25	0>00:00	BR2>14:16	0>00:00	331>15:35	0>00:00							
126>14:45	0>00:00	322>15:15	0>00:00	245>15:00	0>00:00	102>15:00	0>00:00	0>16:00	0>00:00							
242>15:25	0>00:00	209>15:35	0>00:00	366>15:35	0>00:00	144>15:15	0>00:00	0>00:00	0>00:00							
0>16:00	0>00:00	479>15:50	0>00:00	247>15:45	0>00:00	514>15:25	0>00:00	0>00:00	0>00:00							
0>00:00	0>00:00	0>16:00	0>00:00	0>16:00	0>00:00	0>16:00	0>00:00	0>00:00	0>00:00							
0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00							
4<1<0<1<68=71>8>0>0 DIST =319.281752 LATE=1992.982444 LOAD=0.000000 TOTAL=2222.084229 } 602																
AVERAGE DISTANCE=12.445267 SCHEDULE AVERAGE=6.527383 SECONDS TO BUILD=ACTUAL LEGS=165																
STATUS = DONE																

FIG. 6

FIG. 6

UNTITLED RSIM												
FILE EDIT RUN VIEW HELP												
0>00:00	0>16:00	0>08:00	0>16:08	0>08:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>08:00	0>16:00	0>16:00
472>06:11	379>16:28	313>08:25	231>16:18	341>08:15	529>16:08	375>08:17	338>16:10	224>08:27	117>16:28	224>08:27	117>16:28	117>16:28
423>08:31	226>16:48	462>08:51	112>16:41	352>08:28	487>16:27	512>08:31	243>16:32	156>09:03	109>16:50	156>09:03	109>16:50	109>16:50
533>08:48	BR1>17:00	BR1>09:01	483>16:59	194>08:56	587>16:44	509>08:54	164>16:52	134>09:37	202>17:21	134>09:37	202>17:21	202>17:21
566>09:10	311>18:00	596>09:28	382>17:14	364>09:27	484>17:06	373>09:11	319>17:21	BR1>09:47	BR1>17:31	BR1>09:47	BR1>17:31	BR1>17:31
137>09:36	377>18:11	146>10:04	BR1>17:38	BR1>09:40	553>17:31	BR1>09:21	BR1>17:35	127>10:04	468>17:56	127>10:04	468>17:56	468>17:56
BR1>09:48	300>18:36	291>10:15	296>17:57	400>10:12	BR1>17:41	100>09:57	556>17:58	439>10:27	402>18:16	439>10:27	402>18:16	402>18:16
595>10:06	250>19:01	457>10:42	253>18:15	363>10:29	479>18:02	467>10:32	295>18:16	131>10:54	592>18:47	295>18:16	131>10:54	592>18:47
525>10:29	519>19:16	176>11:03	430>18:27	520>10:44	297>18:46	167>11:05	163>18:48	374>11:40	548>19:18	163>18:48	374>11:40	548>19:18
390>10:56	109>19:41	LN>11:13	372>18:49	554>10:59	213>18:58	354>11:17	259>19:06	511>12:05	376>19:46	354>11:17	259>19:06	376>19:46
394>11:30	190>20:05	384>11:58	143>18:19	454>11:37	443>19:12	248>11:47	171>20:00	217>12:36	337>20:05	248>11:47	171>20:00	337>20:05
300>11:51	LN>20:15	572>12:10	335>19:55	410>11:58	475>19:31	536>12:04	LN>20:10	LN>12:50	LN>20:15	536>12:04	LN>20:10	LN>20:15
446>12:02	BR2>22:00	516>12:32	LN>20:16	210>12:10	168>19:48	252>12:23	BR2>22:00	232>13:22	BR2>22:00	252>12:23	232>13:22	BR2>22:00
LN>12:12	0>00:00	501>13:01	BR2>22:00	148>12:46	220>20:08	409>12:00	0>00:00	161>13:46	0>00:00	409>12:00	0>00:00	0>00:00
425>13:06	0>00:00	547>13:18	0>00:00	LN>12:50	LN>20:16	LN>13:00	0>00:00	296>14:17	0>00:00	LN>13:00	0>00:00	0>00:00
104>13:25	0>00:00	263>13:34	0>00:00	153>13:31	BR2>22:00	106>13:33	0>00:00	BR2>14:27	0>00:00	106>13:33	0>00:00	0>00:00
422>13:45	0>00:00	444>13:57	0>00:00	542>13:58	0>00:00	346>13:58	0>00:00	258>14:52	0>00:00	346>13:58	0>00:00	0>00:00
158>14:12	0>00:00	BR2>14:35	0>00:00	428>14:15	0>00:00	236>14:06	0>00:00	239>15:26	0>00:00	236>14:06	0>00:00	0>00:00
BR2>14:22	0>00:00	392>14:59	0>00:00	BR2>14:25	0>00:00	BR2>14:18	0>00:00	331>15:39	0>00:00	BR2>14:18	0>00:00	0>00:00
126>14:47	0>00:00	322>15:18	0>00:00	245>14:52	0>00:00	102>15:00	0>00:00	0>00:00	0>00:00	102>15:00	0>00:00	0>00:00
242>15:24	0>00:00	209>15:37	0>00:00	366>15:17	0>00:00	144>15:14	0>00:00	0>00:00	0>00:00	144>15:14	0>00:00	0>00:00
0>00:00	0>00:00	479>15:48	0>00:00	247>15:39	0>00:00	514>15:26	0>00:00	0>00:00	0>00:00	514>15:26	0>00:00	0>00:00
0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00
0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00	0>00:00
4<1<0<1<68=71>8>8>0>0 DIST =319.281752 LATE=1992.982444 LOAD=0.000000 TOTAL=2222.084229												
AVERAGE DISTANCE=12.445267 SCHEDULE AVERAGE=6.527383 SECONDS TO BUILD=ACTUAL LEGS=165												
STATUS = DONE												

6/8

FIG. 7

7/8

ADD DELIVERY TO SCHEDULE X

802a	CUST ID	42	<input type="radio"/> MOVING <input type="radio"/> AFTERNOON <input type="radio"/> EVENING <input type="radio"/> LATE NIGHT <input checked="" type="radio"/> APP. TIME SET TIME SLOTS	SELECT A TIME SLOT			
802b	VALUE	20		<div style="border: 1px solid black; height: 15px; width: 100%;"></div>			
802c	F BINS	0		14:45 \$6.25 L3 TO 12:00 \$6.25 L2 TO 13:25 \$6.25 L3 TO			
802d	C BINS	0		804a	804b	804c	804d
802e	D BINS	2					
802f	SP TIME	10					

TYPE 1 803a

805 **CANCEL** **OK**

801

FIG. 8

REMOVE DELIVERY X

CUSTOMER ID TO REMOVE 902

42

REMOVE ALL RESERVATIONS

REMOVE ALL GHOSTS 903

904

905 **CANCEL** **OK**

FIG. 9

8/8

TRUCK: 0		1006	1001	1002	1003	1005	1000				
LEG	CUST ID	TARGET	ACTUAL	DIST	TIME	SERV	VALUE	F BINS	C BINS	D BINS	
0	0	08:00	08:00	0.000	00:00	00:00	0.00	0	0	0	
1	472	08:15	08:11	5.466	00:11	00:10	20.00	0	0	2	
2	423	08:35	08:31	3.236	00:06	00:10	20.00	0	0	2	
3	533	08:48	08:48	1.907	00:03	00:10	20.00	0	0	2	
4	565	09:10	09:10	2.609	00:05	00:10	20.00	0	0	2	
5	137	09:35	09:38	9.104	00:18	00:10	20.00	0	0	2	
6	BRK1	00:00	09:48	0.000	00:00	00:15	0.00	0	0	0	
7	595	10:05	10:06	1.245	00:02	00:10	20.00	0	0	2	
8	525	10:30	10:29	6.723	00:13	00:10	20.00	0	0	2	
9	390	10:55	10:56	8.134	00:16	00:10	20.00	0	0	2	
10	394	11:35	11:30	12.233	00:24	00:10	20.00	0	0	2	
11	300	11:50	11:51	3.042	00:06	00:10	20.00	0	0	2	
12	446	12:00	12:02	0.856	00:01	00:10	20.00	0	0	2	
13	LNCH	00:00	12:12	0.000	00:00	00:30	0.00	0	0	0	
14	425	13:10	13:06	11.882	00:23	00:10	20.00	0	0	2	
15	104	13:20	13:25	2.587	00:05	00:10	20.00	0	0	2	
16	422	13:55	13:45	5.267	00:10	00:10	20.00	0	0	2	
17	150	14:10	14:12	3.725	00:07	00:10	20.00	0	0	2	
18	BRK2	00:00	14:22	0.000	00:00	00:15	0.00	0	0	0	
19	126	14:45	14:47	4.899	00:09	00:10	20.00	0	0	2	
20	242	15:25	15:24	13.480	00:27	00:10	20.00	0	0	2	
21	0	16:00	15:58	11.983	00:24	00:00	0.00	0	0	0	

FIG. 10

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US01/07507

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) : G06F 17/60

US CL : 705/8

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 705/7-9

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EAST (US Patent Database; JPO Abstract Database); DIALOG

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,623,413 A (MATHESON et al) 22 April 1997 (22.04.1997), see at least column 5, lines 58-64, column 6, lines 36-42, 46-51, column 9, lines 13-15.	1-31
X	US 5,265,006 A (ASTHANA et al) 23 November 1993 (23.11.1993), see at least column 16, lines 46-59, section "Load Planning" beginning on column 23.	1-31
A	JP 05-313583 (MATSUSHITA ELECTRIC IND CO LTD) 26 November 1993 (26.11.1993), see at least the Abstract.	1-31

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

Special categories of cited documents:	
"A" document defining the general state of the art which is not considered to be of particular relevance	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"E" earlier application or patent published on or after the international filing date	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

16 July 2001 (16.07.2001)

Date of mailing of the international search report

02 AUG 2001

Name and mailing address of the ISA/US

Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231

Facsimile No. (703)305-3230

Authorized officer

Tariq Hafiz

Telephone No. (703)305-3900

James R. Matthews

Form PCT/ISA/210 (second sheet) (July 1998)

1

THIS PAGE BLANK (USPTO)